

Exercise 1: JavaScript Debugging

Problem Statement:

You've been given a simple JavaScript code snippet that's intended to toggle the visibility of an element when a button is clicked. However, it's not working as expected.

Issue: The issue was related to the `toggleElement()` function. The original code used a ternary operator that did not switch between the "block" and "none" values as expected, resulting in the element not toggling correctly.

Correction: Modified the `toggleElement()` function to use an if-else statement for proper toggling of the display property.

Code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Toggle Element</title>
  </head>
  <body>
    <button onclick="toggleElement()">Toggle
Element</button>
    <div id="target" style="display: none">This is the
target element.</div>

    <script>
      function toggleElement() {
        var element = document.getElementById("target");
        if (element.style.display === "none") {
          element.style.display = "block";
        } else {
          element.style.display = "none";
        }
      }
    </script>
  </body>
</html>
```

Exercise 2: CSS Troubleshooting

Issue: The issue was that the container was not properly centered on the page. The margin property was not being applied correctly, which affected the centering of the container.

Correction: Adjusted the CSS properties to ensure the container is centered on the page.

Additionally, added the text-align property to center the text within the container.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Centered Container</title>
    <style>
      .container {
        width: 50%;
        margin: auto;
        background-color: #f0f0f0;
        padding: 20px;
        text-align: center; /* Added for centering text */
      }
    </style>
  </head>
  <body>
    <div class="container">
      <h1>Centered Container</h1>
      <p>This container should be centered on the page.</p>
    </div>
  </body>
</html>
```

Exercise 3: Debugging JavaScript Functions

Objective: Identify and fix issues in JavaScript functions.

This code snippet with a JavaScript function that performs a specific task, but contains bugs or inefficiencies.

Debug the function and ensure it works correctly and efficiently.

Corrections:

The function `calculateSum(arr)` is a simple function that takes an array of numbers as a parameter and returns the sum of its elements. The code snippet you provided has no syntax errors, but it can be improved in terms of performance and readability.

One possible improvement is to use the built-in array method `reduce()` instead of a `for` loop to calculate the sum. The `reduce()` method takes a callback function that applies an operation to each element of the array and accumulates the result in an accumulator variable. This way, you can avoid declaring and incrementing a loop variable `i`, and simplify the code.

Another possible improvement is to use `const` instead of `let` to declare the `sum` variable, since it is not reassigned in the function. This can prevent accidental reassignment or modification of the variable, and make the code more consistent.

Code:

```
function calculateSum(arr) {  
  const sum = arr.reduce((acc, cur) => acc + cur, 0);  
  return sum;  
}  
  
const numbers = [1, 2, 3, 4, 5];  
const result = calculateSum(numbers);  
console.log(result); // Should output 15
```

Exercise 4: Debugging CSS Styling Issues

Issue: The issue was that the boxes within the container were not properly aligned and styled. The alignment was not correct, and the boxes were not evenly distributed within the container.

Correction: Added `display: flex` to the container to enable the use of flexbox for better alignment and distribution of the boxes. Adjusted the `justify-content` property to ensure even spacing between the boxes.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Styling Debugging Exercise</title>
    <style>
      .container {
        width: 50%;
        margin: 0 auto;
        background-color: #f0f0f0;
        padding: 20px;
        display: flex; /* Added to enable flexbox */
        justify-content: space-around; /* Adjusted to
distribute boxes evenly */
      }

      .box {
        width: 100px;
        height: 100px;
        background-color: #007bff;
        color: #ffffff;
        text-align: center;
        line-height: 100px;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <div class="box">Box 1</div>
      <div class="box">Box 2</div>
      <div class="box">Box 3</div>
    </div>
  </body>
</html>
```